

Advanced JavaScript Tracking and Analytics Solution

Bruno Mota

17 Julho, 2014

Prof. João Cardoso **Orientador**
Filipe Silva **Supervisor**



Índice

Introdução	Contextualização
	Caracterização do problema
	Objetivos
Trabalho realizado e implementação	Estado da arte
	Implementação
Testes e conclusões	Testes e resultados
	Conclusões e trabalho futuro

Tracking

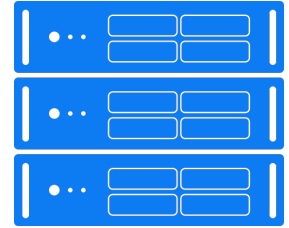


CLIENTE

Recolhe, mede e comunica dados



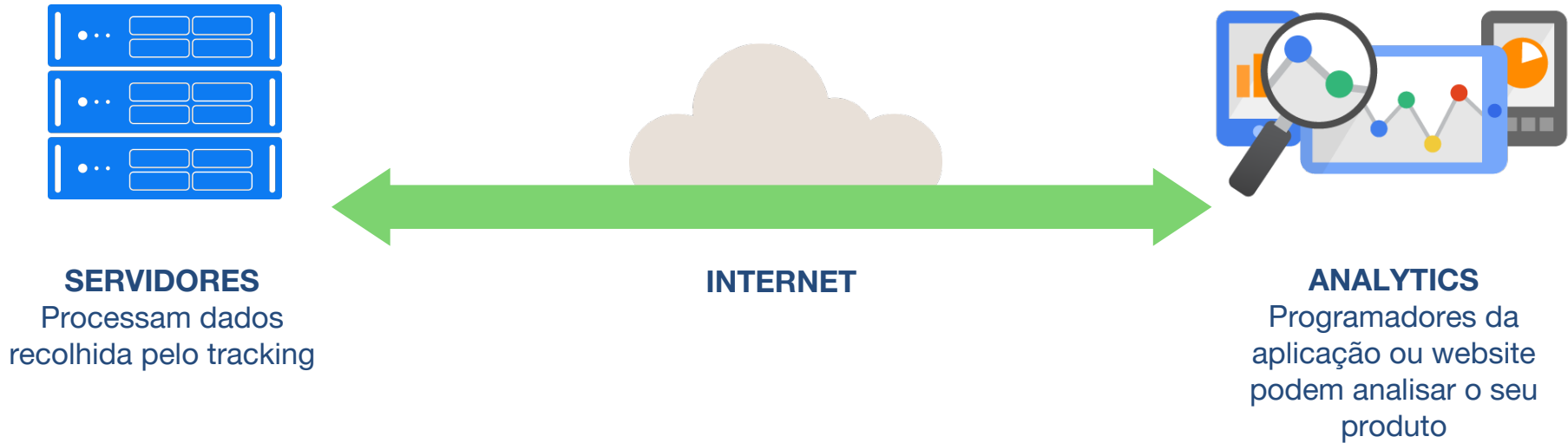
INTERNET



SERVIDORES

Armazenam os dados

Analytics



Web Analytics - Problema

Browser do Cliente



Servidor website

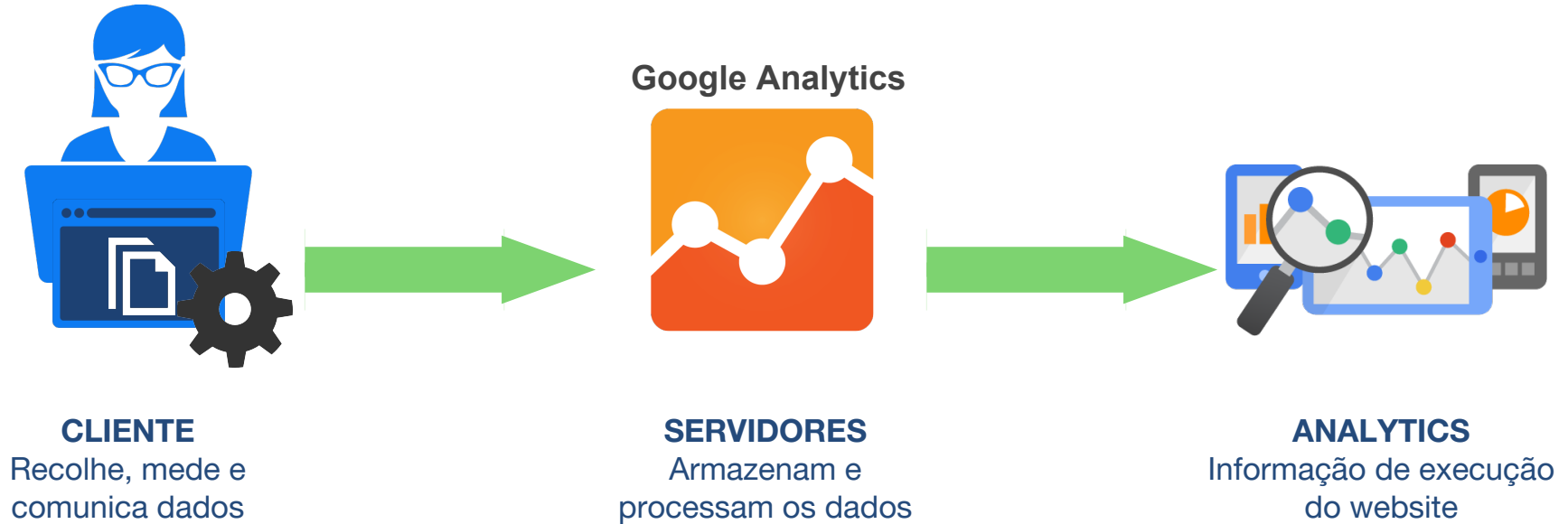


GET

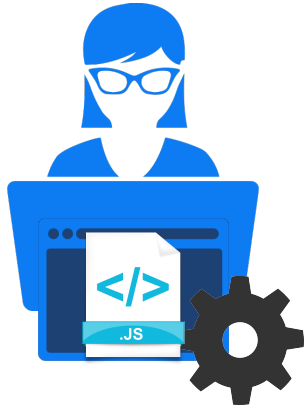
files

Nenhuma informação de
interação e do utilizador

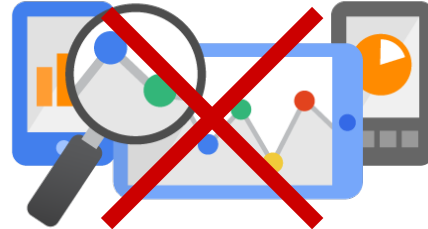
Web Analytics - Google Analytics



JavaScript Analytics - Problema



CLIENTE
JavaScript executado
pelo browser do
cliente



ANALYTICS
Nenhuma informação de
execução do código

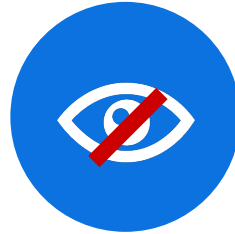
Porquê JavaScript Analytics?



**POPULARIDADE DA
LINGUAGEM**



**DIFERENTES MOTORES
JAVASCRIPT**



**NENHUM FEEDBACK DE
EXECUÇÃO**



POUCO EXPLORADA

Proposta de solução: JStrace

Programador




Ofuscado e com
funcionalidades de
tracking

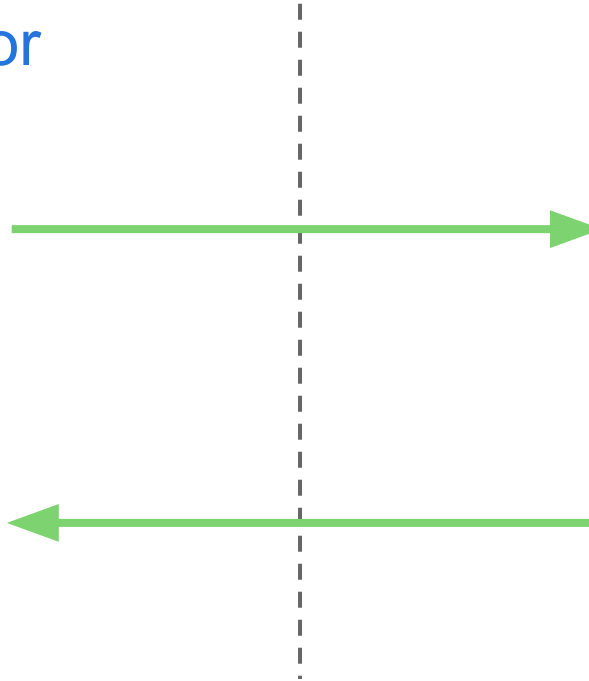


JStrace

Aplicação registada no
sistema



 **jsCrambler**
Instrumentado
automaticamente



Proposta de solução: JStrace

Browser do Cliente



Programador



JStrace

Dados de
execução

REPORT





PERFIL DOS UTILIZADORES

- Execuções
- Localização
- Browser
- Sistema Operativo



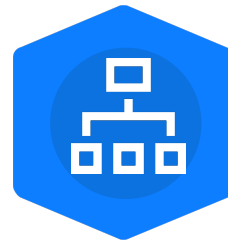
ERROS

Recolha de erros



PERFORMANCE

Cálculo de
performance de
cada função



FLUXO (trace)

Fluxo de execução
do código JavaScript



AUTOMÁTICO

Processo automático de instrumentação do código JavaScript



POUCO INVASIVO

As técnicas de tracking inseridas não devem prejudicar em demasia a performance da aplicação



PROTEGIDO

Uso de ofuscação de modo a que o código de tracking não possa ser retirado



- Grandes quantidades de dados
- Escalável
- Gestão e processamento de dados
- Interface web analytics



Estado da arte - Analytics

	JSTrace	JavaScript Analytics	Application Analytics	Web Analytics
Visitas / Execuções	✓	✗	✓	✓
Perfil dos utilizadores	✓	✗	✓	✓
Tracking de erros	✓	✓	✓	✗
Tracking de eventos	✓	✗	●	●
Performance de funções	✓	✗	●	●
Fluxo de execução	✓	✗	✗	✗
Proteção	✓	✗	✓	✗
Framework exemplo	-	Raygun	Localytics	Google Analytics



Automático



Manual



Inexistente

Estado da arte - JScrambler



Código não protegido



Código protegido

```
(function() {  
  function add1(element, index, array) {  
    array[index]=element+1;  
  }  
  function apply(functions, integers) {  
    if(!Array.isArray(functions)) {  
      integers.forEach(functions);  
    } else {  
      for(var i=0; i<functions.length; i++)  
        integers.forEach(functions[i]);  
    } return integers;  
  } return apply([add1,add1], [1,2,-4,9]);  
})();
```

```
/*...*/ ff)<<W);s=this[a](s,B);s=((s&0x1fff` = A)|  
(s>>>Q` <) f);P^=s;P=((P&0x7` H`"y)| (P>>>h);P=(P*z+  
((129.,0x187)<(91,0x1E0)? (0x72,0xe6546b64): (120.7  
E1,0x35)))|R8H.j3H; }s=` ##switch (t%x){case X:s=(J  
[I](H+v)&0xff)<<c;` 8 v:s|` 5$R8H.t3H` ;$e` >|` 0`"  
` ?&` ;";s=this[a](s,B);s=((s&0x1fff` [ A)|(s>>>Q`  
<) f);P^=s; }P^=t;P^=P>>>c;P` :$P,0x85ebca6b` G  
P>>>y` 2*c2b2ae35` 8$c;return P;};};var r=K.T(`"  
3af`)?"`":{`F`: {(49.80E1>(0xAF,13.)?(22.40E1,  
0):65.>(137.8E1,0x209)? (74,0x84): (0x1AF,46` K )},  
E=L>o,G;for (n0N<10N.length;n0N++) /*...*/
```

JStrace, analytics para aplicações JavaScript

Trabalho realizado e implementação / Implementação

The screenshot displays the JStrace application interface. On the left is a dark sidebar with the JStrace logo and navigation links: Video Background, Home, Dashboard, Analytics, and Execution Tree. The main content area is divided into two panels. The left panel, titled 'Execution Tree', shows a tree structure for 'video_background.js' with nodes 'main (100%)', 'null (100%)', and 'make_youtube (48.8%)'. The right panel, titled 'Overview', features a warning message about performance issues and two charts: 'Paths with Errors' and 'Paths w/ Performance Issues'.

JStrace beta version 0.0.0

Execution Tree
Generated in 26-6-2014

Overview

Warning! Your app seems to have some performance issue functions!

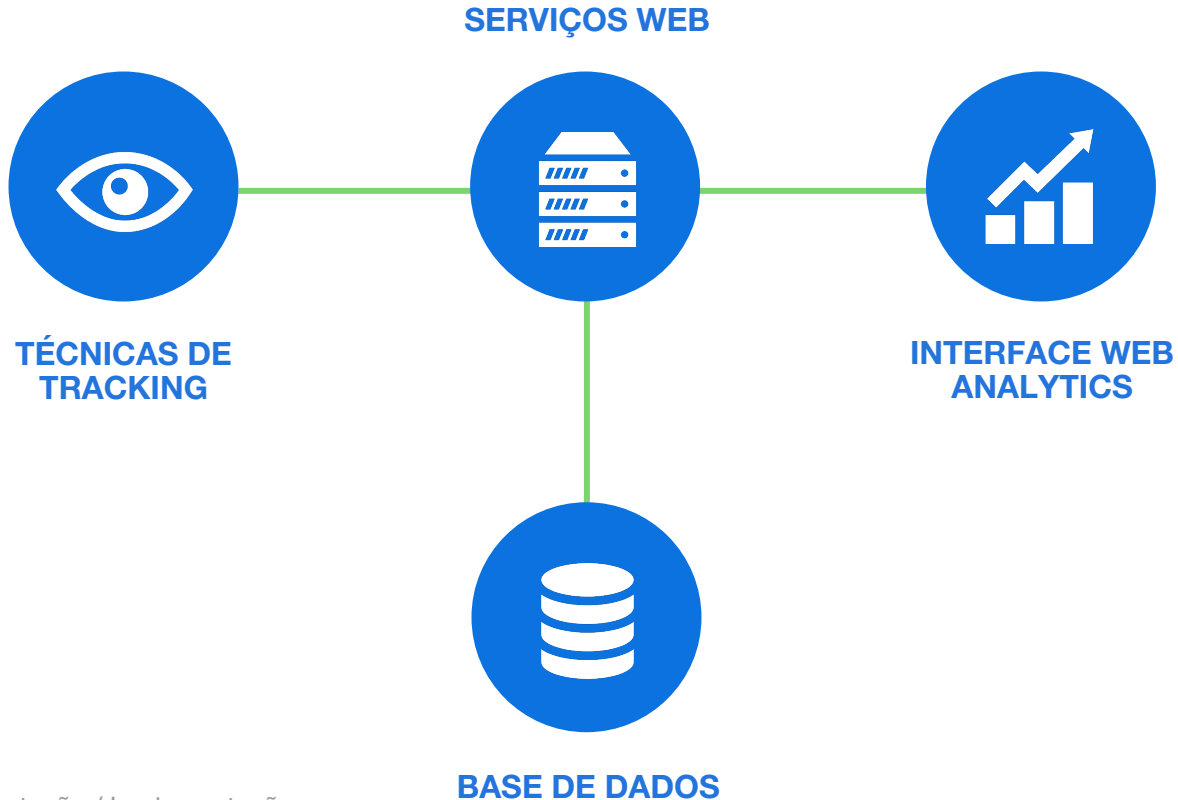
video_background.js

- main (100%)
- null (100%)
 - null (100%)
 - make_youtube (48.8%)

Paths with Errors

Paths w/ Performance Issues

Implementação





Técnicas de tracking

Execuções & Perfil do utilizador

Cada vez que a aplicação
JavaScript é executada

POST id_aplicação



Dados incluídos no
cabeçalho do pedido
processados e
guardados

Apenas uma vez por execução



Técnicas de tracking

Performance & Fluxo de execução

Código original

```
function funcao1() {  
    // código do programador  
    funcao2();  
}  
function funcao2() {  
    // código do programador  
}
```

Código instrumentado

```
function funcao1() {  
    jstrace.start_function("funcao1", 1);  
    // código do programador  
    funcao2();  
    jstrace.end_function(4);  
}  
function funcao2() {  
    jstrace.start_function("funcao2", 5);  
    // código do programador  
    jstrace.end_function(7);  
}
```



Técnicas de tracking

Performance & Fluxo de execução

Relatório de fluxo e performance

Gerado periodicamente durante a execução da aplicação

```
{
  id: "5217a543dd99a6d9e0f74702",
  report: {
    "funcao1": {
      "count": 1,
      "time": 3,
      "order": 0
    },
    "funcao1.funcao2": {
      "count": 1,
      "time": 1,
      "order": 0
    }
  }
}
```

POST relatório





Técnicas de tracking

Erros

Erros previstos

```
try{  
  // código do programador  
}  
catch(err) {  
  jstrace.handled_error(err); // tracking  
}
```

Erros não expectáveis

Apanhados por um evento
`window.onerror`

Relatório enviado para o servidor

```
{  
  type: 1,  
  msg: error,  
  fun: "funcao1.funcao2",  
  time: timeSpent,  
  line: 5  
}
```



Técnicas de tracking

Proteção

Código minificado

```
jstrace.start_function("main","fish.js");var  
ctx,startFish=10,fish=[],fishW=307,fishH=313,  
velocity=100,backgroundImageW=1010,  
backgroundImageH=600,imageStrip,WIDTH=document.  
body.offsetWidth,HEIGHT=document.body.  
offsetHeight,fpsMeter=null;window.onload=init;  
function init(){jstrace.start_function("init","  
fish.js",33);fpsMeter=new FpsMeter(0,"fish");  
fpsMeter.SetSettingsHtml  
(...)
```



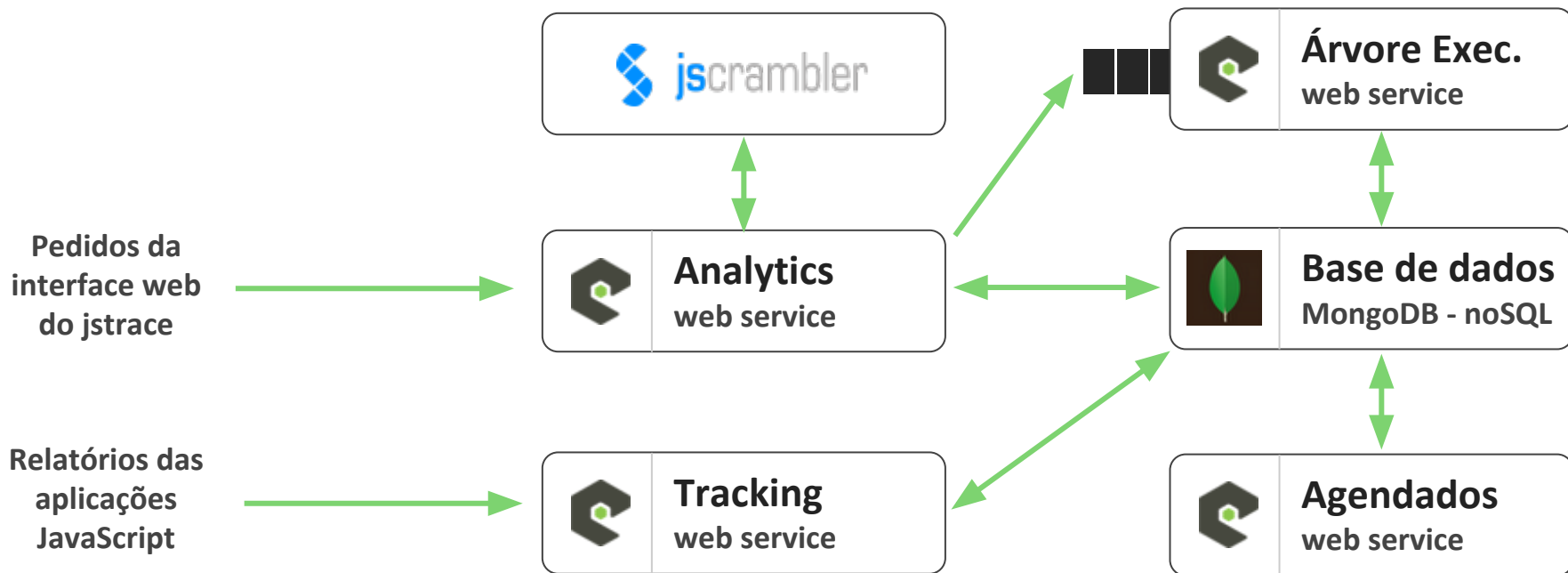
Código ofuscado

```
var U5J=(function D(X,U){var t='',k=unescape  
( 'q%18S-%1C%19%05%08%25%3D%3B+%5E%3F%08f%0C%  
043%13%10/E/i%08%16%0C/5%1F9*%5D%5B9XH%0B%3C%  
1E%0F@%1A%1C%28%23%2CC%0A%167%1C%19%05%0D%3F%  
3D+0%5B5%1D6Z%168%17%11%29I%1B%24%0EX%09-%20%  
1F%21%3AI%1E%3F%12F%146%0B%00%10L%154%04*X%0D%  
1F+LO%274%03%16%17%00w%0C%27%02%3E%27%138%27%  
13r%1B%0C%26%1E%3F%12F%146%0B%00%10L%154%04*X%  
0D%1F+LO%274%03  
(...)
```



Serviços Web

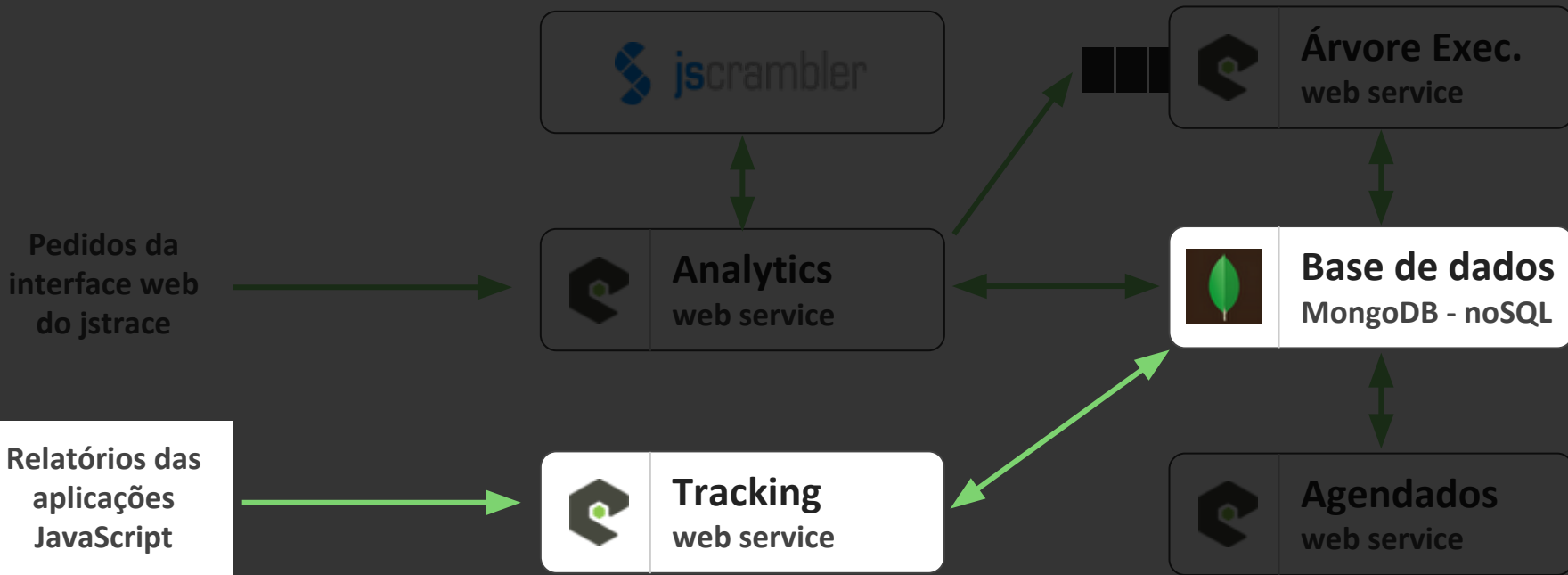
Estrutura dos serviços





Serviços Web

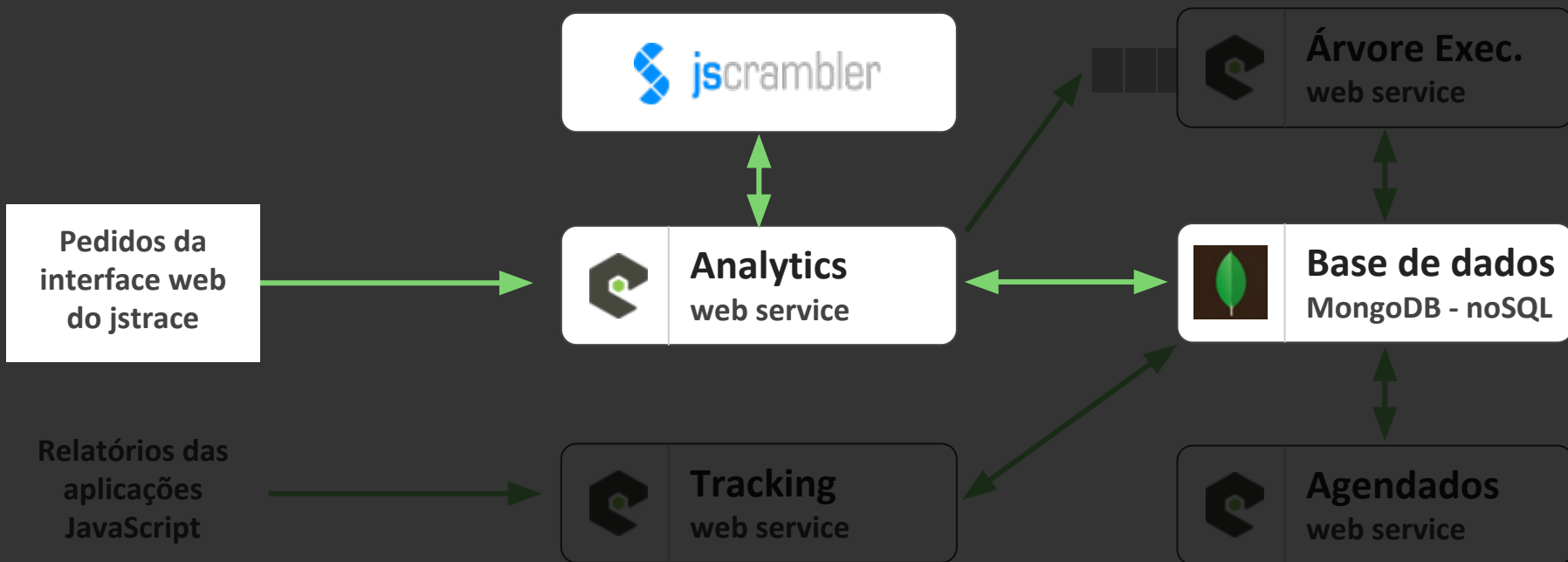
Tracking





Serviços Web

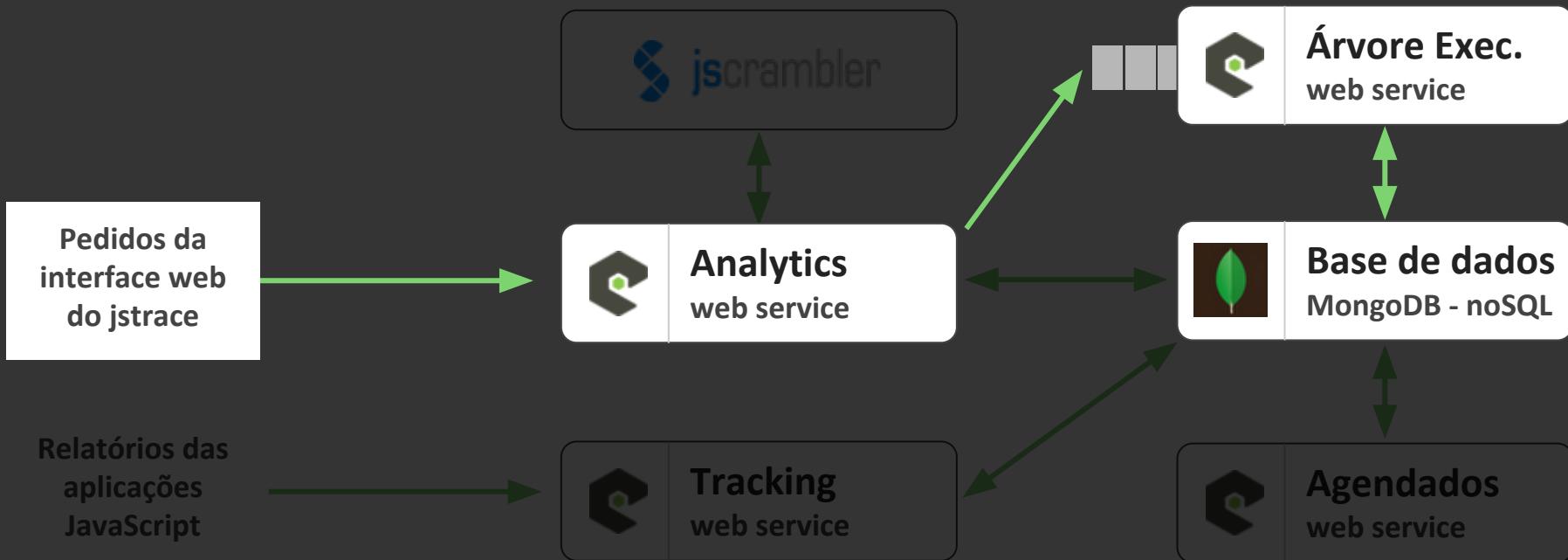
Interface Web





Serviços Web

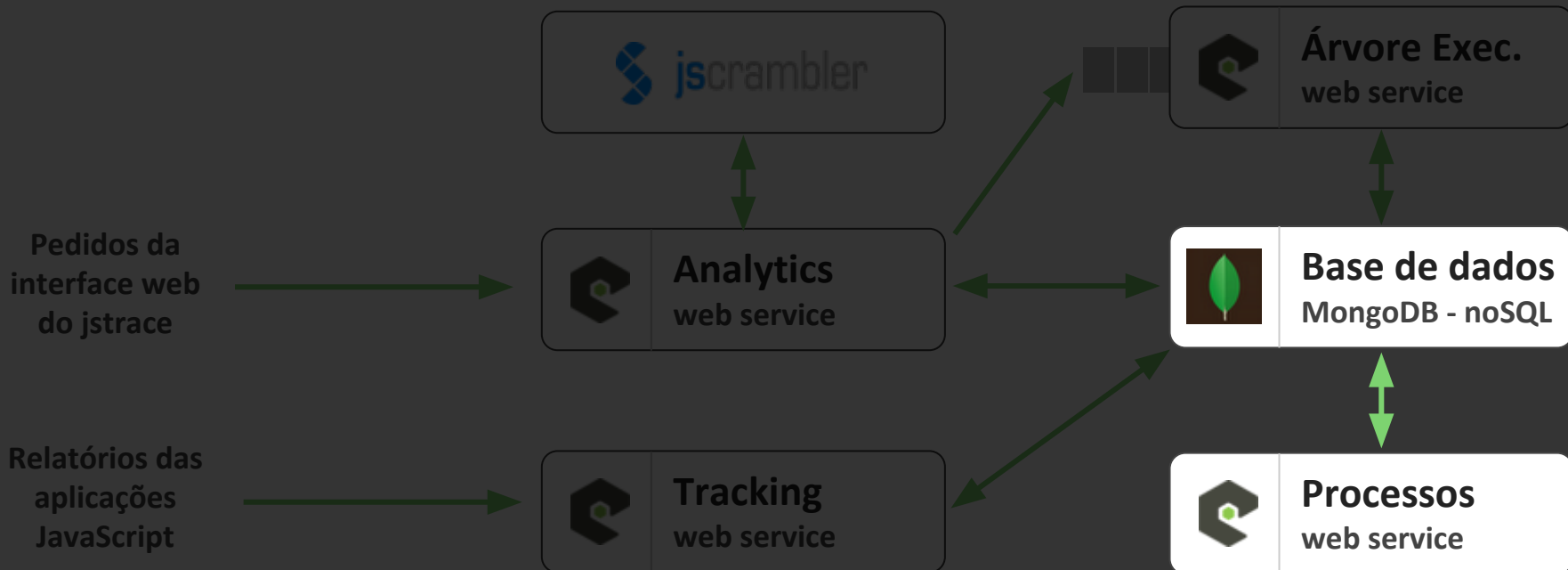
Gerar árvore de exec.





Serviços Web

Processos Assíncronos

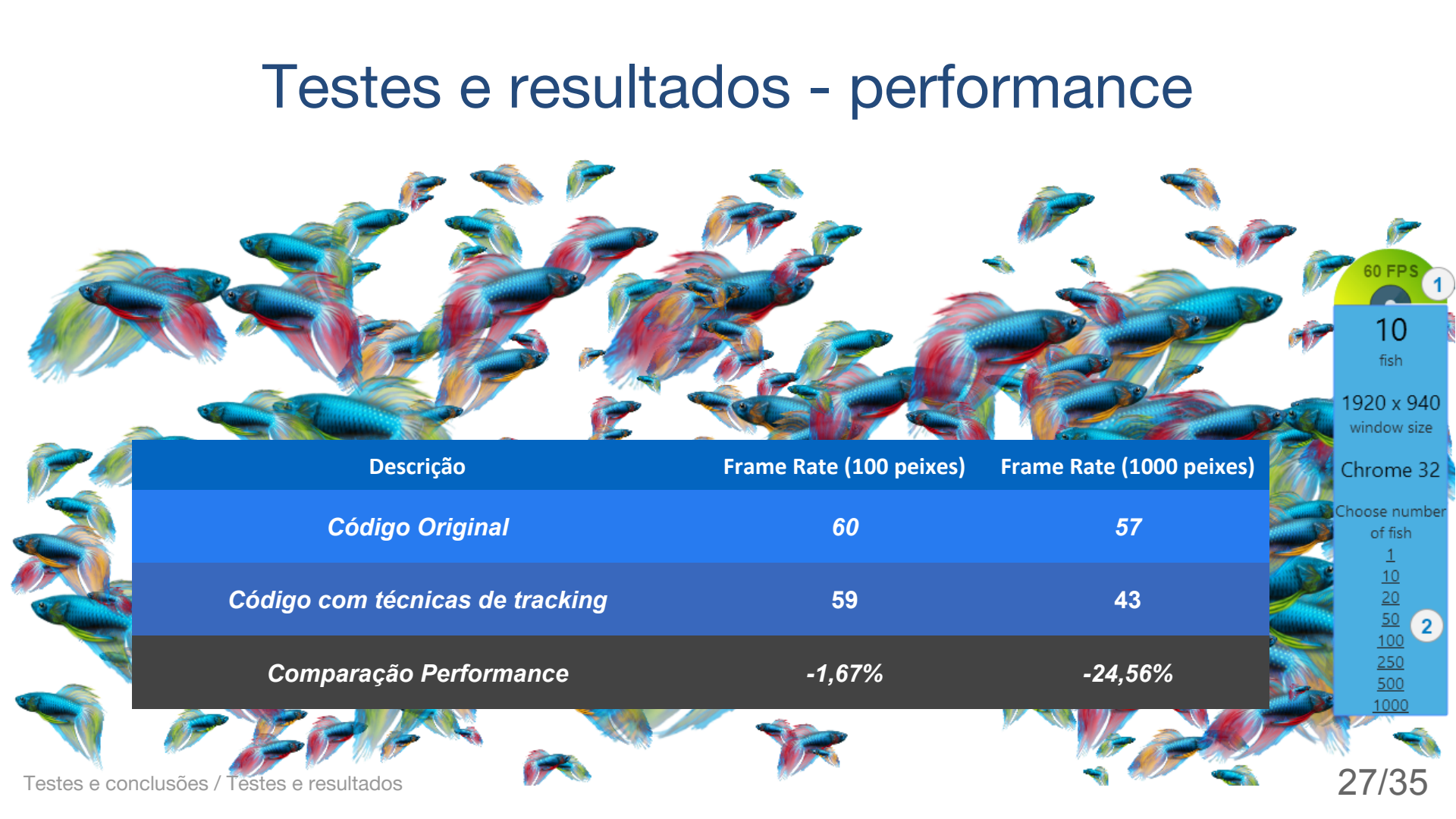




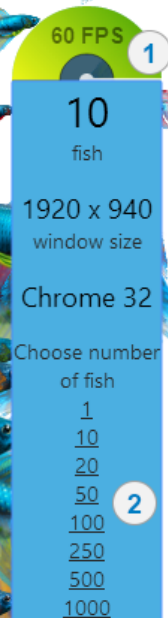
Interface Web Analytics

Onde o programador pode registrar e analisar aplicações

Testes e resultados - performance



Descrição	Frame Rate (100 peixes)	Frame Rate (1000 peixes)
<i>Código Original</i>	60	57
<i>Código com técnicas de tracking</i>	59	43
<i>Comparação Performance</i>	-1,67%	-24,56%



60 FPS 1

10
fish

1920 x 940
window size

Chrome 32

Choose number
of fish

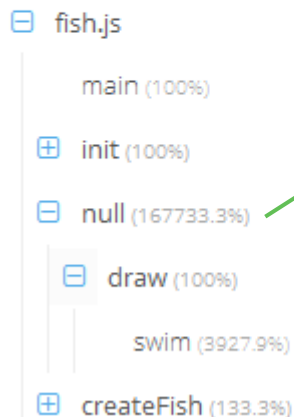
1
10
20
50
100
250
500
1000

2

Testes e resultados - performance

Execution Tree

Generated in 3-7-2014



Executada 1677 vezes em média.

Executada $1677 \times 39 = 65403$ vezes em média.

Testes e resultados - performance

Descrição	Frame Rate (100 peixes)	Frame Rate (1000 peixes)
<i>Código Original</i>	60	57
<i>Código com técnicas de tracking (alterado manualmente)</i>	60	56
<i>Comparação Performance</i>	~ 0%	-1,75%

Testes e resultados - caso de uso

Código minificado

```
jstrace.start_function("main","fish.js");var
ctx,startFish=10,fish=[],fishW=307,fishH=313,
velocity=100,backgroundImageW=1010,
backgroundImageH=600,imageStrip,WIDTH=document.
body.offsetWidth,HEIGHT=document.body.
offsetHeight,fpsMeter=null;window.onload=init;
function init(){jstrace.start_function("init","
fish.js",33);fpsMeter=new FpsMeter(0,"fish");
fpsMeter.SetSettingsHtml
(...)
```

Código ofuscado

```
jstrace[("start"+"_f"+"u"+"n"+"ction")](("ma"+"
in"),("fi"+"sh"+"."+"j"+"s"));var U5J=(function
D(X,U){var t=' ',k=unescape('q%18S-%1C%19%05%08%
25%3D%3B+%5E%3F%08f%0C%043%13%10/E/i%08%16%
0C/5%1F9*%5D%5B9XH%0B%3C%1E%0F@%1A%1C%28%23%
2CC%0A%167%1C%19%05%0D%3F%3D+0%5B5%1D6Z%168%17%
11%29I%1B%24%0EX%09-%20%1F%21%3AI%1E%3F%12F%
146%0B%00%10L%154%04*X%0D%1F+LO%274%03%16%17%
00w%0C%27%02%3E%27%138%27%13r%1B%0C%26%
(...)
```


Testes e resultados - caso de uso

Código minificado

Performance by browser

Browser	# Runs	Avg time(ms)	Min time(ms)	Max time(ms)	Deviation(ms)
Firefox	1	4	4	4	✓ 1
Chrome	1	9	9	9	^ 4
IE	1	2	2	2	✓ 3

Função “main”

Performance by browser

Browser	# Runs	Avg time(ms)	Min time(ms)	Max time(ms)	Deviation(ms)
Firefox	1	87	87	87	✓ 16
Chrome	1	100	100	100	✓ 3
IE	1	122	122	122	^ 19

Código obfuscado

Testes e resultados - caso de uso

Código minificado

Performance by browser

Browser	# Runs	Avg time(ms)	Min time(ms)	Max time(ms)	Deviation(ms)
Firefox	1	15	15	15	▼ 2
Chrome	1	13	13	13	▼ 4
IE	1	23	23	23	▲ 6

Função “init”

Performance by browser

Browser	# Runs	Avg time(ms)	Min time(ms)	Max time(ms)	Deviation(ms)
Firefox	1	15	15	15	▼ 15
Chrome	1	25	25	25	▼ 5
IE	1	51	51	51	▲ 21

Código obfuscado

Conclusões



- ✓ **Comprovado analytics para JavaScript**
- ✓ **Controlo de erros**
- ✓ **Fluxo de execução**
- ✓ **Performance**
- ✓ **Dados dos utilizadores**
- ✓ **Proteção**
- ✓ **Sistema de analytics completo**

Trabalho Futuro



- ✓ Técnicas de tracking inline
- ✓ Transferir processamento do cliente para o servidor
- ✓ *Stress test* ao sistema
- ✓ Lançar versão beta
- ✓ Funcionalidade de remover o código de tracking em determinadas funções através da interface web

Perguntas?

